# Dijkstra Worksheet

Name:
NetID:
Signature:

1. What is the defining characteristic of a Max Heap? How can the characteristics of Heaps be used to implement a Priority Queue?

Ans: Parents must have larger values than their children, and the root node has the biggest value. In a Min Heap, parents must always have a smaller value than the child, so the root node is always the node with the smallest value.

2. Suppose we have an array representation of a Binary Heap. Find the parent, left child, and right child of an element $i$.

Ans: Parent → Floor((i-1)/2)
Left Child → 2*i+1
Right Child → 2*i+2

3. How can you insert into a Heap? Assume the Heap is implemented as an array. Write some code!

```
def HeapInsert(Heap,key):
  Heap.size = Heap.size + 1
  i = Heap.size
  while ((i > 0) and (Heap[parent(i)] > key)):
    Heap[i] = Heap[parent(i)]
    i = parent(i)

  Heap[i] = key
```

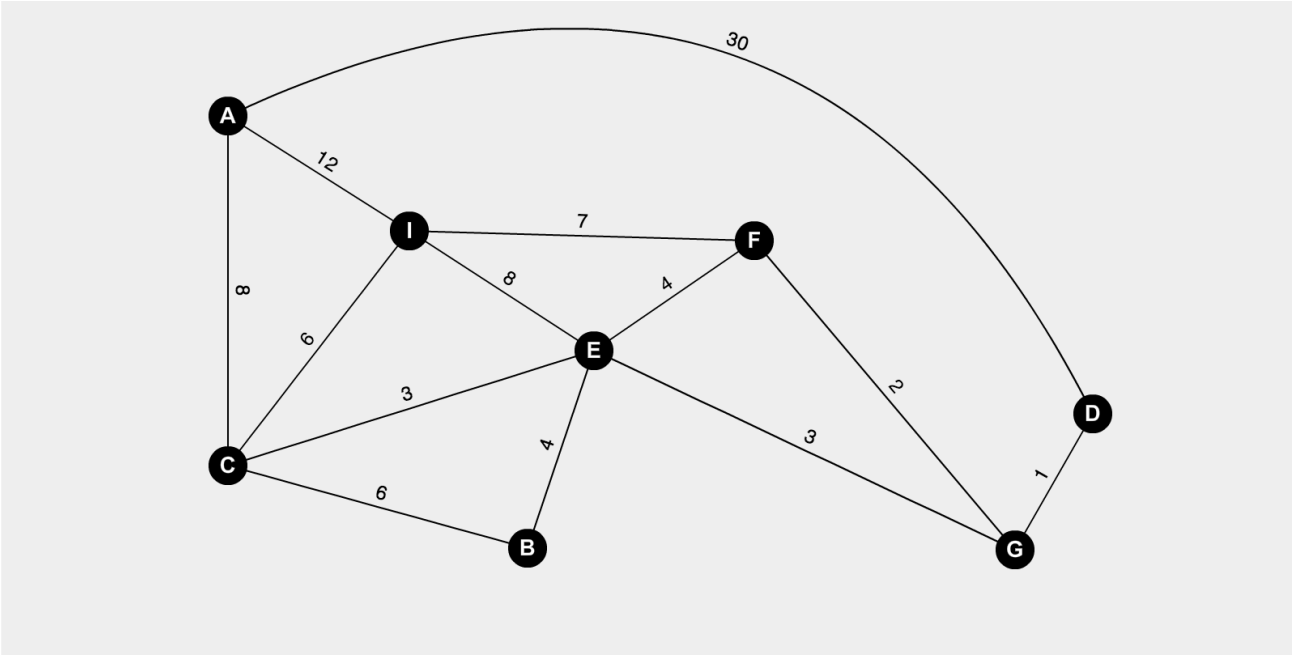
4. Let's say you have a very important packet to send from your computer to another computer, find the shortest path from your computer to all other nodes in the computer network.

Here is Dijkstra:

```
def Dijkstra(Graph, start, end):
  PQ = Queue.PriorityQueue()
```

```
  foreach node in Graph:
    node.cost = infinity
  start.cost = 0
  foreach node in Graph:
    PQ.insert(n,node.cost)
  while not(PQ.empty()):
    u = PQ.extract()
    if u == end:
      break
    foreach neighbor v in u.neighbors:
      w = cost from v to u
      newCost = u.cost + w
      if (newCost < v.cost):
        PQ.decreasekey(v,newcost)
        v.cost = newCost
        v.predecessor = u
```

Here is a graph. Use Dijkstra to find the shortest path from A to all nodes in the graph. Go through iteration by iteration, update the costs, as well as $u$ from the psuedocode. $\infty$



| | $u$ | A | B | C | D | E | F | G | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | A | 0 | inf | 8 | 30 | inf | inf | inf | 12 |
| 2 | AC | 0 | 14 | 8 | 30 | 11 | inf | inf | 12 |
| 3 | ACE | 0 | 14 | 8 | 30 | 11 | 15 | 14 | 12 |
| 4 | ACEI | 0 | 14 | 8 | 30 | 11 | 15 | 14 | 12 |
| 5 | ACEIG | 0 | 14 | 8 | 15 | 11 | 15 | 14 | 12 |
| 6 | ACEIGB | 0 | 14 | 8 | 15 | 11 | 15 | 14 | 12 |